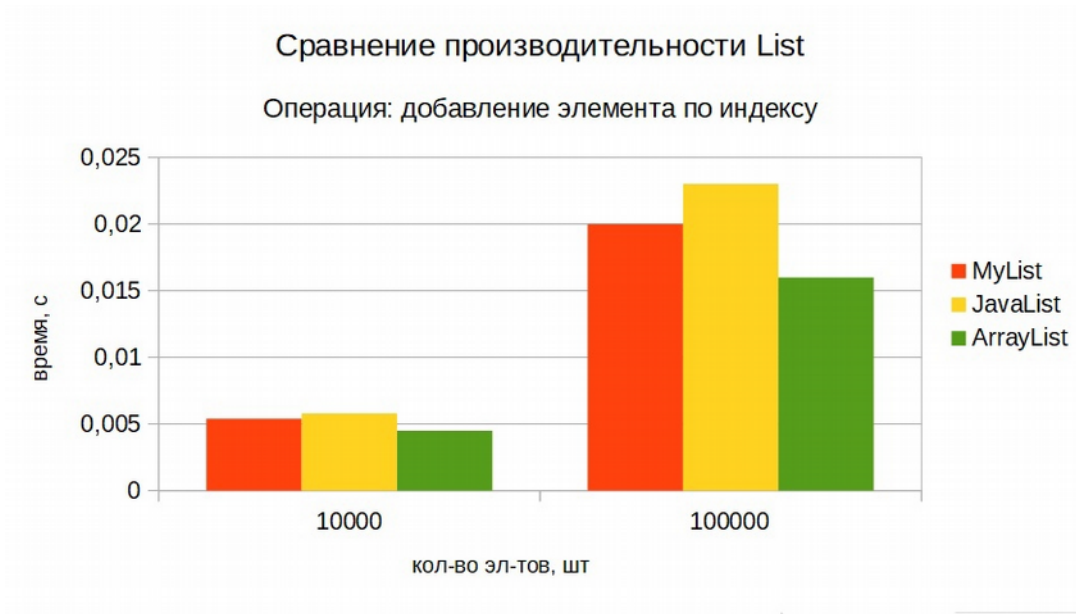


Сравнение производительности коллекций для основных операций

Используя System.nanoTime оценим примерно время выполнения основных операций.

1. MyLinkedList, LinkedList, ArrayList





Подытоживая полученные данные, имеем следующее:

ArrayList эффективнее для основных операций, чем LinkedList(так как в этих тестах индекс выбирается случайным образом и данных было достаточно много, то есть создавались ситуации, в которых как LinkedList, так и ArrayList были максимально неэффективны).

Рассмотрим случаи, когда эффективнее использовать ArrayList и LinkedList.

Когда эффективнее использовать LinkedList:

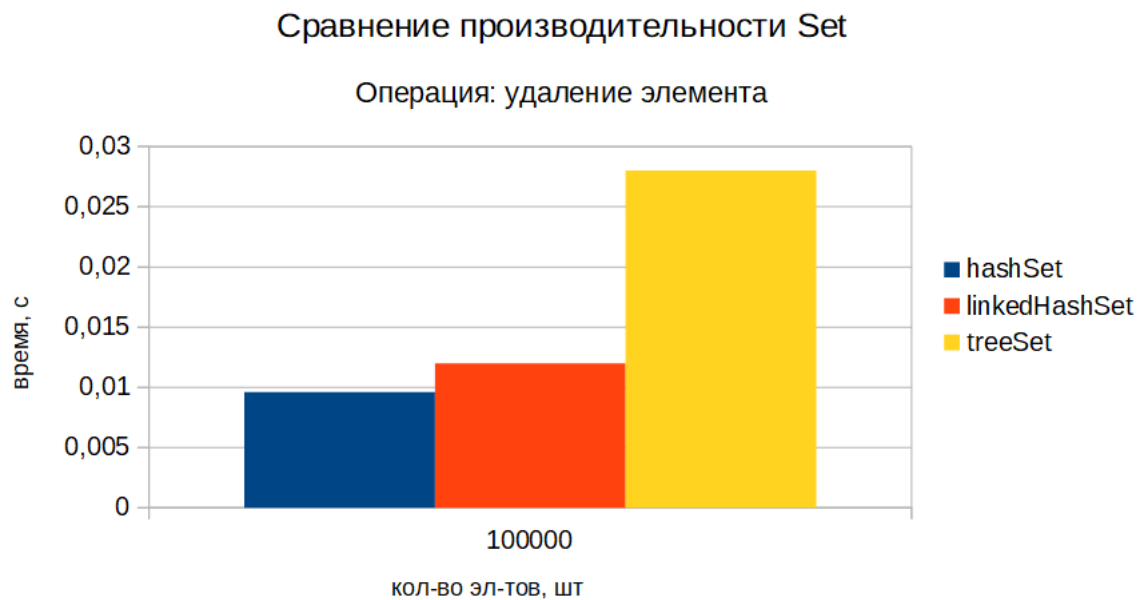
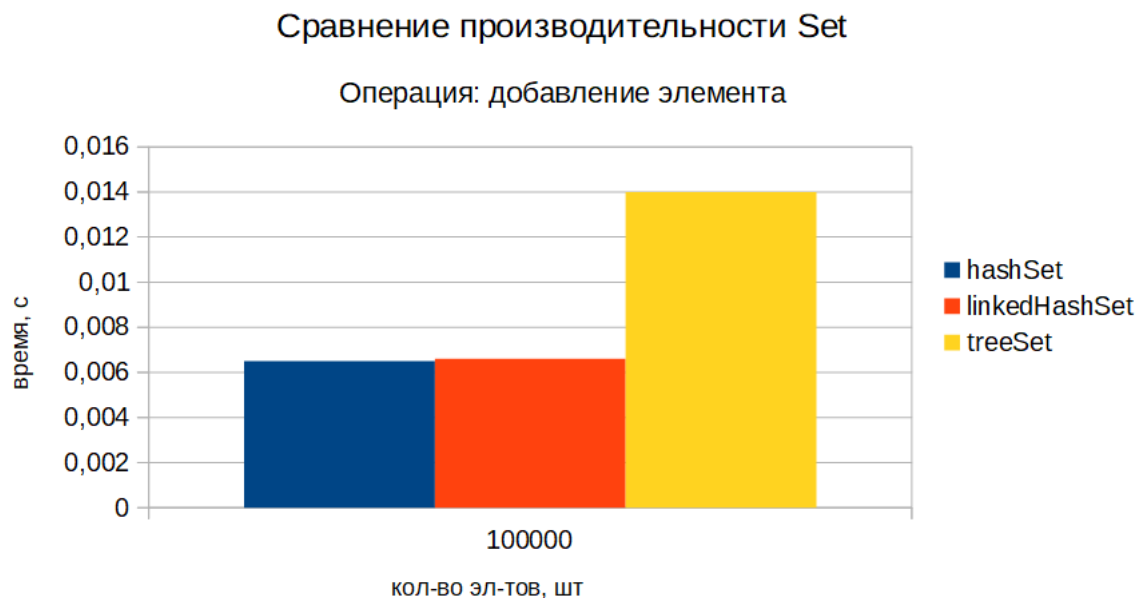
1. Необходимо много данных добавлять по индексу в начало и конец списка.
2. Удалять, извлекать с начала и с конца списка(если индекс будет располагаться ближе к середине, LinkedList будет неэффективен).

Когда эффективнее использовать ArrayList:

1. Получить элемент по индексу.
2. Вставка, удаление элемента (только если индекс ближе к концу массива, так как при вставке/удалении происходит перекопирование массива и ArrayList будет неэффективен при вставке/удалении из начала массива).

Стоит отметить, что LinkedList стабильнее добавляет элементы, так как ArrayList нужно расширяться.

2. HashSet, LinkedHashSet, TreeSet



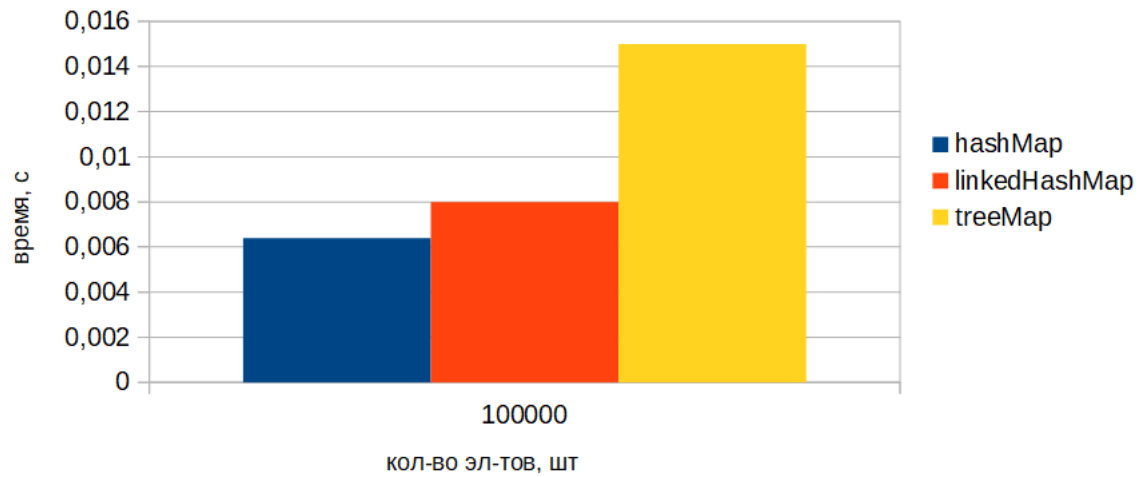
Set представляет собой множество без повторяющихся элементов. HashSet, LinkedHashSet, TreeSet отличаются между собой способом хранения данных.

Если нам нужна отсортированная коллекция, то используем TreeSet, если нужна коллекция, которая может хранить порядок вставки, используем LinkedHashSet. В остальных случаях используем HashSet.

3. HashMap, LinkedHashMap, TreeMap

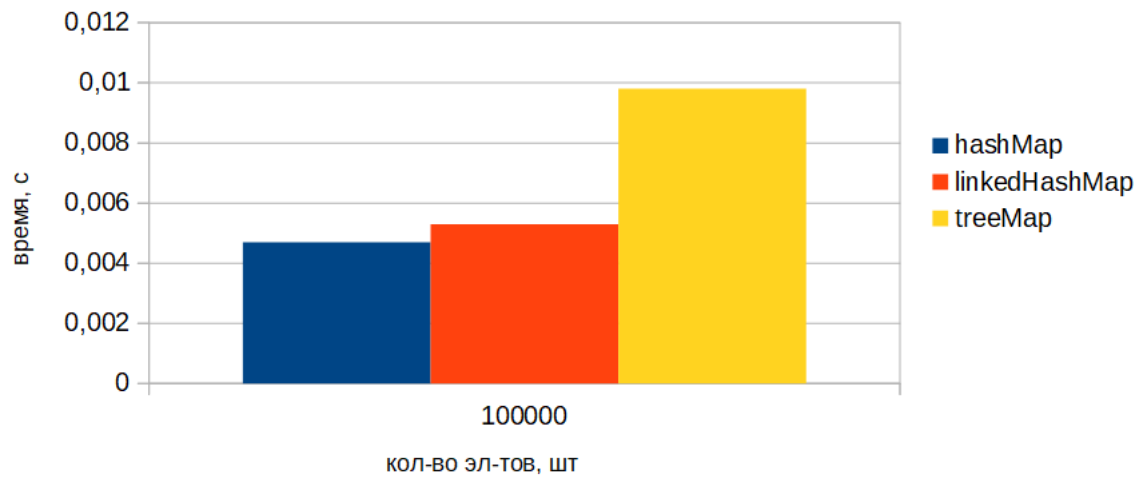
Сравнение производительности Map

Операция: вставка элемента



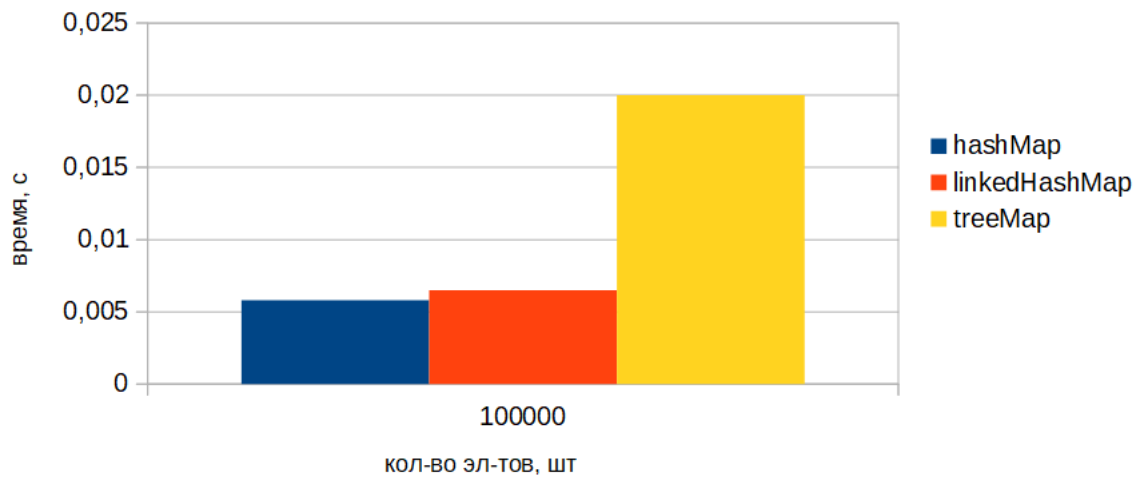
Сравнение производительности Map

Операция: получение элемента



Сравнение производительности Map

Операция: удаление элемента



Map – структура, которая хранит пару (ключ, значение), где ключ уникальный. HashMap, LinkedHashMap, TreeMap отличаются между собой способом хранения данных.

Если нам нужна отсортированная коллекция, то используем TreeMap, если нужна коллекция, которая может хранить порядок вставки, используем LinkedHashMap. В остальных случаях используем HashMap.